In the Claims

The status of claims in the case is as follows:

1. [Currently amended] Method for processing a client session request received at a server in a system including a client, a server, and ~~a legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held on a connection for duration of a dialogue, comprising the steps of:

said client connecting to said server;

said client and said server negotiating environment parameters for establishing a connection-oriented connection of said server with said client, said client and said server communicating over said connection using a same client/server communications protocol, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

while negotiating said environment parameters, said server inviting said client to negotiate terminal type and submit user environment variables;

said client responding by returning to said server said terminal type and submitting a request for a custom confirmation record, said request including at least

one user variable;

responsive to receiving ~~a user~~ said user variable and said request for ~~requesting~~ a custom confirmation record ~~received at said server~~ from said client, said server executing an exit program for calling and passing said user variable to a host application at said host external to said server, said host application processing said user variable and responsive thereto returning custom data to said server, said custom data selectively including a user variable received from said client that was selected and used; and

said server concluding negotiating said environment parameters with said client selectively including sending to said client a confirmation record ~~and custom record~~ including said custom data received from said exit program ~~; for enabling said client to engage in subsequent programmable negotiations directly with said server~~.

2. [Original] The method of claim 1, said negotiating, inviting, and sending steps executing within the application layer of a TCP/IP protocol stack.

3. [Currently amended] The method of claim 1, further comprising the step responsive to a user variable requesting a confirmation record, sending to said client a confirmation record without said custom ~~record~~ data.

4. [Original] The method of claim 1, said confirmation

record including a field defining a pass through data length, said pass through data including said confirmation record and said custom data.

5. [Currently amended] The method of claim 1, further comprising the step of appending said custom ~~record~~ data to said confirmation record.

6. [Currently amended] The method of ~~claim a1~~ claim 1, said request being for a default custom confirmation record, and further comprising the step of sending to said client default data received at said exit program at said server from said host application in said custom ~~record~~ data.

7. [Currently amended] The method of claim 1, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, further comprising the step of sending to said client defined data in said custom ~~record~~ data.

8. [Currently amended] The method of claim 7, said sending step including executing at said server a customer defined exit program on said list to access said host to generate said defined data.

9. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data received at said exit program at said server from said host application indicia identifying a device allocated by a host server.

10. [Currently amended] The method of claim 4, further

comprising the step of providing in said custom ~~record~~ data indicia received at said exit program at said server from said host application identifying a terminal or printer device allocated by ~~a host server~~ said host.

11. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data indicia received at said exit program at said server from said host application identifying an associated device linked to a current session by a host.

12. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data indicia received at said exit program at said server from said host application identifying a physical location for receiving output.

13. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data indicia received at said exit program at said server from said host application identifying system security level and password encryption requirements.

14. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data indicia identifying another device for retrying a rejected request.

15. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data indicia identifying a reason for a failed auto-signon request.

16. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

17. [Currently amended] The method of claim 4, further comprising the step of providing in said custom ~~record~~ data indicia <u>received at said exit program at said server from said host application</u> identifying custom information for interpretation by said client.

18. [Currently amended] A client/server system including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ <u>said server</u> executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, comprising:

a custom confirmation record;

a user exit program running on said server;

said client operating in conjunction with said user exit program for requesting said custom confirmation record from said server, and responsive thereto for engaging in subsequent client/server negotiations; said client and said server communicating over a connection-oriented connection using a same client/server communications protocol, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an

application layer with said server;

a host application program module for receiving from said exit program a user variable provided to said server by a client request for a custom confirmation record and responsive thereto for returning to said server custom data selectively including said user variable;

said server further for sending to said client a confirmation record including said custom data.

19. [Original] The system of claim 18, said client being a Telnet client.

20. [Currently amended] The system of claim 18, further comprising:

said client being selectively operable for negotiating a send-custom-confirmation-record with a 'yes', 'no' or defined data value; and

said user exit ~~interpret~~ interpreting said data value and sending default or defined information received at said exit program at said server from said host application back to said client in said custom confirmation record.

21. [Currently amended] The system of claim 20, said custom confirmation record containing diagnostic information provided by said server along with custom information

~~provided~~ received at said exit program at said server from said host application by said user exit program.

22. [Currently amended] The system of claim 21, said custom information being provided by user exit programs executing in said server ~~and said client~~ to call application programs at said host.

23. [Currently amended] A method for operating a client to establish a network connection with a server in a system including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, comprising the steps of:

said client connecting to said server;

said client negotiating with said server environment parameters for establishing a connection-oriented connection with said server, said client and said server communicating over said connection using a same client/server communications protocol, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

said client receiving from said server an invitation to negotiate terminal type and submit user environment variables;

said client responding to said invitation by requesting ~~parameters including a request for~~ said server to provide a custom confirmation record, the request including at least one user variable; and

~~responsive to said request,~~ receiving at said client said custom confirmation record ~~at said client and engaging in subsequent programmable negotiations directly with said server~~, said custom confirmation record received at said client including custom data provided by a host application program responsive to receiving said user variable from an exit program executing at said server.

24. [Currently amended] The method of claim 23, said custom confirmation record including return code, system name, device name and said custom data.

25. [Original] The method of claim 24, further comprising the steps of:

operating said server to request a custom information record from said client.

26. [Original] The method of claim 25, said request comprising an invitation to said client from said server to respond with all environment variables.

27. [Original] The method of claim 26, said client responding to said invitation by returning a custom information record as part of said environment variables.

28. [Original] The method of claim 27, said client responding to said invitation with a request that said server return to said client a custom confirmation record.

29. [Currently amended] The method of claim 28, further the steps of

operating an exit program at said server to call an application at said host to interpret the value in said custom information record to selectively return a custom confirmation record response.

30. [Currently amended] The method of ~~claim 28~~ claim 29, further comprising the steps of specifying in said custom confirmation record a list of custom fields to be returned by said server.

31. [Currently amended] The method of claim 28, further comprising the steps of specifying in said custom confirmation record unstructured data for subsequent parsing and processing by said server, an application program at said host called by an exit program at said server, or an independent job.

32. [Currently amended] Method for operating a client to establish a network connection with a server in a system including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, comprising the steps of:

said client connecting to said server;

said client negotiating with said server environment parameters for establishing a connection-oriented connection with said server, said client and said server communicating over said connection using a same client/server communications protocol, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

while negotiating said environment parameters, receiving from said server an invitation to negotiate terminal type and submit user environment variables;

said client responding by returning to said server said terminal type and submitting a request for a custom confirmation record, said request including at least one user variable;

responsive to sending to said server ~~a user~~ said user variable requesting a custom confirmation record, receiving at said client from said server a confirmation record and custom record data for enabling said client to engage in subsequent negotiations directly with said server, said custom record data generated by said host responsive to execution of a server exit program passing to a host application said user variable.

33. [Original] The method of claim 32, said negotiating, inviting, and sending steps executing within

the application layer of a TCP/IP protocol stack.

34. [Currently amended] The method of claim 32, further comprising the step, responsive to said invitation to submit user variables, of requesting a confirmation record, and responsive thereto receiving from said server a confirmation record without said custom record data.

35. [Original] The method of claim 32, said confirmation record including a field defining a pass through data length, said pass through data including said confirmation record and said custom record data.

36. [Original] The method of claim 32, further comprising the step of receiving said custom record data appended to said confirmation record.

37. [Original] The method of claim 32, said request being for a default custom confirmation record, and further comprising the step of receiving from said server, default data in said custom record data.

38. [Currently amended] The method of claim 32, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, further comprising the step of receiving from said server, client defined data provided by a host application responsive a server exit program in said custom record data.

39. [Currently amended] The method of claim 38, further including the step of providing to said server a customer

defined exit program accessing a host application program for parsing said list to generate said defined data.

40. [Currently amended] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a device allocated by ~~a host server~~ said host application.

41. [Currently amended] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a terminal or printer device allocated by ~~a host server~~ said host application.

42. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying an associated device linked to a current session by a host.

43. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a physical location for receiving output.

44. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying system security level and password encryption requirements.

45. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying another device for retrying a rejected request.

46. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a reason for a failed auto-signon request.

47. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

48. [Original] The method of claim 35, further comprising the step of receiving in said custom record data indicia identifying custom information for interpretation by said client.

49. [Currently amended] A client system for establishing a network connection with a server in a system including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, comprising:

a first logic element for negotiating environment parameters for establishing a connection-oriented connection with said server;

said parameters including a request for said server to provide a custom confirmation record to said client, said request including at least one user variable, said client including a graphical user interface selectively

assigned a session name enabling client emulator communication at an application layer with said server; and

a second logic element ~~responsive to said request,~~ for receiving said confirmation record from said server, said confirmation record including custom data provided to an exit program at said server by a host application external to said server for enabling said client to engage in subsequent programmable negotiations with said server, said client and said server communicating over said connection using a same client/server communications protocol.

50. [Original] The system of claim 49, said custom confirmation record including return code, system name, device name and custom data.

51. [Original] The system of claim 50, further comprising:

a third logic element for operating said server to request a custom information record from said client.

52. [Original] The system of claim 51, said request comprising an invitation to said client from said server to respond with all environment variables.

53. [Original] The system of claim 52, said client further comprising a fourth logic element for responding to said invitation by returning a custom information record as part of said environment variables.

54. [Original] The system of claim 53, said client further comprising a fifth logic element for responding to said invitation with a request that said server return to said client a custom confirmation record.

55. [Currently amended] The system of claim 54, said server further comprising an exit program for calling an application at said host for interpreting the value in said custom information record to selectively return a custom confirmation record response.

56. [Original] The system of claim 54, further comprising a logic element for specifying a list of custom fields to be returned by said server in said custom confirmation record.

57. [Currently amended] The system of claim 54, further comprising a logic element for specifying in said custom confirmation record unstructured data for subsequent parsing and processing by said server, an application at said host called by said exit program, or an independent job.

58. [Currently amended] System including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs on a session connection request for processing a client session request in which direct communication between said client and said server is held for duration of a dialogue, comprising:

a logic element at said server for negotiating environment parameters for establishing a connection-oriented connection with said client and inviting said

client to negotiate terminal type and submit user variables to said server, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server; and

a logic element at said client for returning to said server said terminal type and a request for a custom confirmation record, said request including at least one user variable; and

an exit program at said server, responsive to receiving a user variable from said client requesting a custom confirmation record, for executing an exit program requesting of an application program at said host custom data for sending to said client in a confirmation record ~~and custom record data for enabling said client to engage in subsequent programmable negotiations directly with said server, said client and said server communicating over said connection using a same client/server communications protocol~~.

59. [Original] The system of claim 58, further comprising a TCP/IP protocol stack including within an application layer said exit program generating said custom record data.

60. [Original] The system of claim 58, said logic element further operable responsive to a user variable requesting a confirmation record for sending to said client a confirmation record without said custom record data.

61. [Original] The system of claim 58, said confirmation record including a field defining a pass through data length, said pass through data including said confirmation record and said custom record data.

62. [Original] The system of claim 58, said logic element further operable for appending said custom record data to said confirmation record.

63. [Currently amended] System for operating a client to establish a network connection with a server in a system including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, comprising:

a first logic element for connecting to said server and negotiating environment parameters for establishing a connection-oriented connection with said server;

a second logic element [[and]] for receiving from said server an invitation to negotiate terminal type and submit user variables, said client and said server communicating over said connection using a same client/server communications protocol, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

a third logic element at said client for ~~a second logic~~

~~element responsive to~~ sending to said server said terminal type and submitting a request for a custom confirmation record, said request including at least one ~~a user~~ user variable; and

a fourth logic element ~~requesting a custom confirmation record~~ for receiving from said server a confirmation record and custom record data, ~~for enabling said client to engage in subsequent programmable negotiations directly with said server~~ said custom record data generated by a host application selecting and using said user variable passed to said host by an exit program at said server.

64. [Original] The system of claim 63, further comprising a TCP/IP protocol stack including an application layer within which said logic elements execute.

65. [Original] The system of claim 63, further comprising the step responsive to said invitation to submit user variables, requesting a confirmation record, and responsive thereto receiving from said server a confirmation record without said custom record data.

66. [Original] The system of claim 63, said confirmation record including a field defining a pass through data length, said pass through data including said confirmation record and said custom record data.

67. [Original] The system of claim 63, said second logic element further responsive for receiving said custom record data appended to said confirmation record.

68. [Original] The system of claim 63, said request being for a default custom confirmation record, and said second logic element further operable for receiving from said server default data in said custom record data.

69. [Original] The system of claim 63, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, said second logic element further operable for receiving from said server client defined data in said custom record data.

70. [Currently amended] The system of claim 69, further including a logic element for providing to said server a customer defined exit program for calling an application at said host for parsing said list to generate said defined data.

71. [Currently amended] A physical program storage device readable by a machine, tangibly embodying a program of instructions executable by a machine to perform method steps for processing a client session request received at a server in a system including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, said method steps comprising:

said client connecting to said server;

said client and said server negotiating environment

parameters for establishing a connection-oriented connection with said client, said client and said server communicating over said connection using a same client/server communications protocol, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

while negotiating said environment parameters, said server inviting said client to negotiate terminal type and submit user environment variables to said server;

said client responding by returning to said by returning to said server said terminal tyhpe and submitting a request for a custom confirmation record, said request including at least one user variable;

responsive to receiving at said server ~~a user~~ said user variable ~~requesting~~ and said request for a custom confirmation record, said server executing an exit program for calling and passing said user variable to a host application at said host external to said server, said host application processing said user variable and responsive thereto returning custom data to said server, said custom data selectively including a user variable received from ~~sending to~~ said client ~~a confirmation record and custom record data enabling said client to engage in subsequent programmable negotiations directly with said server~~ that was selected and used; and

said server concluding negotiating said environment

paramters with said client selectively including send to said client a confirmation record including said custom data received from said exit program.

72. [Original] The program storage device of claim 71, said negotiating, inviting, and sending steps executing within the application layer of a TCP/IP protocol stack.

73. [Original] The program storage device of claim 71, said method steps further comprising, responsive to a user variable requesting a confirmation record, sending to said client a confirmation record without said custom record data.

74. [Original] The program storage device of claim 71, said confirmation record including a field defining a pass through data length, said pass through data including said confirmation record and said custom record data.

75. [Original] The program storage device of claim 71, said method steps further comprising the step of appending said custom record data to said confirmation record.

76. [Original] The program storage device of claim 71, said request being for a default custom confirmation record, and said method steps further comprising the step of sending to said client default data in said custom record data.

77. [Original] The program storage device of claim 71, said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, and said method steps further comprising

the step of sending to said client defined data in said custom record data.

78. [Original] The program storage device of claim 77, said sending step including executing a customer defined exit program on said list to generate said defined data.

79. [Currently amended] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a device allocated by a host ~~server~~.

80. [Currently amended] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a terminal or printer device allocated by a host ~~server~~.

81. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying an associated device linked to a current session by a host.

82. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a physical location for receiving output.

83. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying system security level and password encryption requirements.

84. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying another device for retrying a rejected request.

85. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a reason for a failed auto-signon request.

86. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

87. [Original] The program storage device of claim 74, said method steps further comprising the step of providing in said custom record data indicia identifying custom information for interpretation by said client.

88. [Currently amended] A physical program storage device readable by a machine, tangibly embodying a program of instructions executable by a machine to perform method steps for operating a client to establish a network connection with a server in a system including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, said method steps comprising:

said client connecting to said server;

said client and said server negotiating environment parameters for establishing a connection-oriented connection of said client with said server, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

receiving at said client from said server an invitation to negotiate terminal type and submit user environment variables, said client and said server communicating over said connection using a same client/server communications protocol;

said client responding by returning to said server said terminal type and submitting a request for a custom confirmation record, said request selectively including a user variable;

said server executing an exit program for calling and passing said user variable to a host application at said host external to said server, said host application processing said user variable and responsive thereto returning custom data to said server, said custom data selectively including a user variable received from said client that was selected and used by said host application; and

~~responsive to sending to said server a user variable requesting a custom confirmation record,~~ receiving at said client from said server a confirmation record ~~and~~

~~custom record~~ including said custom data ~~enabling said client to engage in subsequent programmable negotiations directly with said server~~.

89. [Original] The program storage device of claim 88, said negotiating, inviting, and sending steps executing within the application layer of a TCP/IP protocol stack.

90. [Original] The program storage device of claim 88, said method steps further comprising the step, responsive to said invitation to submit user variables, of requesting a confirmation record, and responsive thereto receiving from said server a confirmation record without said custom record data.

91. [Original] The program storage device of claim 88, said confirmation record including a field defining a pass through data length, said pass through data including said confirmation record and said custom record data.

92. [Original] The program storage device of claim 88, said method steps further comprising the step of receiving said custom record data appended to said confirmation record.

93. [Original] The program storage device of claim 88, said request being for a default custom confirmation record, and said method steps further comprising the step of receiving from said server default data in said custom record data.

94. [Original] The program storage device of claim 88,

said request being for a defined custom confirmation record, said request including a list of one or more predefined information items, said method steps further comprising the step of receiving from said server client defined data in said custom record data.

95. [Original] The method of claim 94, further including the step of providing to said server a customer defined exit program for parsing said list to generate said defined data.

96. [Currently amended] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a device allocated by a host ~~server~~.

97. [Currently amended] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a terminal or printer device allocated by a host ~~server~~.

98. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying an associated device linked to a current session by a host.

99. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a physical location for receiving output.

100. [Original] The program storage device of claim 91,

said method steps further comprising the step of receiving in said custom record data indicia identifying system security level and password encryption requirements.

101. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying another device for retrying a rejected request.

102. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a reason for a failed auto-signon request.

103. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying a reason for denial of session connection request upon system overload and redirection to an alternate time or host.

104. [Original] The program storage device of claim 91, said method steps further comprising the step of receiving in said custom record data indicia identifying custom information for interpretation by said client.

105. [Currently amended] A computer program product embodied on a tangible storage medium for operating a server in a network including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a

dialogue, comprising:

a ~~tangible~~ physical storage medium;

first program instructions for connecting said client to said server;

~~first program~~ second program instructions for said client and said server ~~negotiating~~ to negotiate environment parameters for establishing a connection-oriented connection of said server with a client, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

~~second program~~ third program instructions for ~~inviting~~ said server to invite said client to negotiate terminal type and submit user environment variables to said server, said client and said server communicating over said connection using a same client/server communications protocol;

~~third program~~ fourth program instructions responsive to said server receiving from said client ~~a user variable requesting~~ a request for a custom confirmation record, said request including a user variable, ~~for sending to said client a confirmation record and custom record data enabling said client to engage in subsequent programmable negotiations directly with said server~~ for executing at said server an exit program for calling and passing said user variable to a host application external to said server, said host application

processing said user variable and, responsive thereto, returning custom data to said server and sending to said client from said server a confirmation record including said custom data received from said exit program; and wherein

said first, second, third, ~~and third~~ and fourth program instructions are recorded on said ~~tangible~~ physical storage medium.

106. [Currently amended] A computer program product embodied on a tangible storage medium for operating a client in a network including a client, a server, and a ~~legacy~~ host with ~~both server and client~~ said server executing exit programs for negotiating a confirmation record on a session connection request in which direct communication between said client and said server is held for duration of a dialogue, comprising:

a ~~tangible~~ physical program storage medium;

first program instructions for connecting said client to said server;

second program ~~first program~~ instructions for negotiating environment parameters for establishing a connection-oriented connection of said client with a server, said client including a graphical user interface selectively assigned a session name enabling client emulator communication at an application layer with said server;

third program ~~second program~~ instructions for receiving from said server at said client an invitation to negotiate terminal type and submit user variables, said client and said server communicating over said connection using a same client/server communications protocol;

fourth program ~~third program~~ instructions for returning to said server said terminal type and submitting a request for a custom confirmation record, said request including at least one user variable;

fifth program instructions responsive to ~~sending to~~ said request for executing an exit program at said server for calling and passing said user ~~a user~~ variable to a host application at said host external to said server, said host application processing said user variable and responsive thereto returning custom data to said server, said custom data including a user variable received from said client that was selected and used;

sixth program instructions for concluding negotiation of said environment parameters and ~~requesting a custom confirmation record, for receiving at said client from said server a~~ for providing to said client said confirmation record and custom record data received at said said exit program from said host ~~enabling said client to engage in subsequent programmable negotiations directly with said server~~; and wherein

said first, second, third, fourth, fifth, and sixth ~~and~~

~~third~~ program instructions are recorded on said ~~tangible~~ physical program storage medium.